WHAT IS CLAIMED IS:

1. A processor comprising:

a fetch address generation unit configured to generate a fetch address; and

a line predictor coupled to said fetch address generation unit, said line predictor including a first memory comprising a plurality of entries, each entry storing a plurality of instruction pointers, wherein said line predictor is configured to select a first entry of said plurality of entries, said first entry corresponding to said fetch address, and wherein each of a first plurality of instruction pointers within said first entry, if valid, directly locates an instruction within a plurality of instruction bytes fetched in response to said fetch address.

2. The processor as recited in claim 1 further comprising:

a plurality of decoders configured to decode instructions; and

an alignment unit coupled to receive said plurality of instruction bytes and said first plurality of instruction pointers and further coupled to said plurality of decoders, wherein said alignment unit is configured to align an instruction to each of said plurality of decoders responsive to a corresponding one of said first plurality of instruction pointers.

3. The processor as recited in claim 1 wherein said first entry is further configured to store a next entry indication identifying a second entry of said plurality of entries within said first memory, wherein said line predictor is configured to subsequently select said second entry to provide a second plurality of instruction pointers stored therein

responsive to said next entry indication.

4. The processor as recited in claim 3 further comprising an instruction cache coupled to said line predictor, wherein said next entry indication further includes a next fetch address, and wherein said instruction cache is coupled to receive said next fetch address from said line predictor and to provide a second plurality of instruction bytes in response thereto.

5. The processor as recited in claim 4 wherein said instruction cache is set associative, and wherein said first entry is further configured to store a way prediction corresponding to said next fetch address, and wherein said way prediction identifies which one of a plurality of ways within said instruction cache is to provide said second plurality of instruction bytes.

6. The processor as recited in claim 5 wherein said instruction cache is configured to provide one or more of said second plurality of instruction bytes from a second storage location therein, and wherein said first entry includes a second way prediction corresponding to said second storage location.

7. The processor as recited in claim 1 wherein said first entry is further configured to store control information corresponding to said instructions located by said first plurality of instruction pointers.

8. The processor as recited in claim 7 wherein said control information includes an indication that at least one byte of a last instruction located by said first plurality of instruction pointers is stored on a different page than said plurality of instruction bytes.

9. The processor as recited in claim 8 further comprising a translation lookaside buffer (TLB) configured to translate a second fetch address corresponding to said at least one

byte.

10. The processor as recited in claim 9 wherein said processor is configured to fetch said at least one byte from said different page.

11. The processor as recited in claim 7 further comprising:

an instruction cache configured to store instruction bytes; and

a translation lookaside buffer (TLB) coupled to said instruction cache and configured to translate virtual addresses to physical addresses;

wherein said fetch address is a virtual address, and wherein said TLB is configured to translate said fetch address to a corresponding physical address and to provide said corresponding physical address to said instruction cache to fetch said plurality of instruction bytes.

12. The processor as recited in claim 11 wherein said virtual address comprises a linear address.

13. The processor as recited in claim 3 wherein said line predictor further includes a second memory coupled to receive said fetch address and further coupled to said first memory, said second memory comprising a second plurality of entries configured to store fetch addresses and indexes into said first memory.

14. The processor as recited in claim 13 wherein said second memory is configured to compare said fetch address to fetch addresses stored in said second plurality of entries and to select a second entry of said second plurality of entries in response to said fetch address matching said fetch address stored in said second entry, and wherein said second

memory is configured to provide said index stored in said second entry to said first memory to select said first entry.

15. The processor as recited in claim 14 wherein said line predictor is configured to inhibit access to said second memory if said next entry indication in said first entry is valid.

16. The processor as recited in claim 13 wherein said second memory comprises a content addressable memory (CAM).

17. The processor as recited in claim 16 wherein said first memory comprises a random access memory (RAM).

18. The processor as recited in claim 16 wherein said CAM is configured to compare a portion of said fetch address to said fetch addresses stored in said second plurality of entries.

19. A method comprising:

generating a fetch address; and

selecting a first plurality of instruction pointers from a line predictor, said first plurality of instruction pointers corresponding to said fetch address, each of said first plurality of instruction pointers, if valid, directly locating an instruction within a plurality of instruction bytes fetched in response to said fetch address.

20. The method as recited in claim 19 further comprising aligning each of said instructions within said plurality of instruction bytes to a plurality of decoders in response

to a respective one of said plurality of instruction pointers.

21. The method as recited in claim 19 wherein said line predictor comprises a first memory including a plurality of entries, each of said plurality of entries configured to store a plurality of instruction pointers, and wherein said selecting comprises selecting a first entry of said plurality of entries, said first entry storing said first plurality of instruction pointers.

22. The method as recited in claim 21 wherein said first entry is further configured to store a next entry indication, the method further comprising selecting a second entry of said plurality of entries responsive to said next entry indication.

23. The method as recited in claim 22 wherein said next entry indication includes a next fetch address, the method further comprising:

providing said next fetch address to an instruction cache; and

accessing a first storage location in said instruction cache in response to said next fetch address.

24. The method as recited in claim 23 wherein said instruction cache is set associative, and wherein said first entry is further configured to store a way prediction, the method further comprising selecting one of a plurality of ways of said instruction cache from which to fetch said plurality of instruction bytes in response to said way prediction.

25. The method as recited in claim 24 wherein said first entry is further configured to store a second way prediction, the method further comprising:

accessing a second storage location in said instruction cache in response to said

next fetch address; and

selecting one of said plurality of ways in response to said second way prediction.

26. The method as recited in claim 21 wherein said first entry is further configured to store an indication that a last instruction located by said first plurality of instruction pointers includes at least one byte in a different page, the method further comprising:

generating a second fetch address corresponding to said different page;

translating said second fetch address; and

fetching instruction bytes from said instruction cache using said second fetch address.

27. The method as recited in claim 21 wherein said line predictor further comprises a second memory including a second plurality of entries, each of said second plurality of entries storing a particular fetch address and a corresponding index into said first memory, wherein said selecting comprises:

comparing said fetch address to said particular fetch address stored in each of said second plurality of entries;

selecting said corresponding index from a second entry of said second plurality of entries in response to said comparing; and

selecting said first entry responsive to said corresponding index.

28. The method as recited in claim 27 wherein said comparing comprises comparing a

portion of said fetch address to a corresponding portion of said particular fetch address.

29. The method as recited in claim 19 further comprising fetching said plurality of instructions using a physical address translated from said fetch address, said fetch address being a virtual address.

30. A computer system comprising:

a processor comprising:

a fetch address generation unit configured to generate a fetch address; and

a line predictor coupled to said fetch address generation unit, said line predictor including a first memory comprising a plurality of entries, each entry storing a plurality of instruction pointers, wherein said line predictor is configured to select a first entry of said plurality of entries, said first entry corresponding to said fetch address, and wherein each of a first plurality of instruction pointers within said first entry, if valid, directly locates an instruction within a plurality of instruction bytes fetched in response to said fetch address; and

an input/output (I/O) device configured to communicate between said computer system and another computer system to which said I/O device is couplable.

31. The computer system as recited in claim 30 wherein said I/O device comprises a modem.